# Naïve Bayes Documents

## Multilabel naïve Bayes classification considering label dependence

*Hae-Cheon Kim, Jin-Hyeong Park, Dae-Won Kim, Jaesung Lee∗*

**Good overview of different algorithms**

, the conventional multilabel naïve Bayes classifier treats each label independently and hence neglects the relations among labels, resulting in degenerated accuracy. For example, in the weather classification problem, the label raining is likely to be coupled with the label cloudy and unlikely to be coupled with the label sunny.

In this paper, we propose a new multilabel naïve Bayes algorithm that considers the dependence among labels for the classification process, named MLNB-LD

e label dependence can be broadly divided into three groups according to how many labels are considered concurrently.

1. The first group of classifiers treats each label independently by inferencing a mapping function for each label. For example, Zhang and Zhou.
2. In the second group, multilabel classifiers consider the label dependence between label pairs. For example, Huang et al.
3. In the third group, the classification process is designed to consider an arbitrary number of labels concurrently.

We presented a multilabel naïve Bayes classifier that considers the dependence among labels during classification. The proposed method utilizes the dependence between label pairs for determining the most probable label set for a given unseen instance. Our comprehensive experiments demonstrate that multilabel classification performance can be significantly improved by the proposed method. A comparison of the results obtained on 14 realworld datasets obtained from different domains shows the advantages of the proposed method compared with the four conventional multilabel classifiers in terms of three evaluation measures

## A versatile framework for labelling imagery with a large number of

## Classes

*Shailesh Kumar', Melba Crawford2 and Joydeep Ghoshl; Department of Electrical and Computer Engineering University of Texas at Austin*

**Refers to letter dataset**

A Bayesian pairwise classifier architecture for a large number of classes was proposed and evaluated in this paper. This architecture is novel in three ways. First it provides a framework for pair-class feature selection that was not available in the conventional feature selection techniques. Second, the evaluation criteria uses log odds measure on class aposteriori probabilities to advantage. Third, a fast incremental algorithm is used for learning MOG’s in higher dimensional feature space using the MOG’s in the lower dimensional feature subspaces, based on knowledge reuse methods.

# Random Forest Documents, Decision Trees

## Random Decision Forests

*Tin Kam Ho AT&T Bell Laboratories[1]*

**Compares decision trees and random forests**

But trees derived with traditional methods often cannot be grown to arbitrary complexity for possible loss of generalization accuracy on unseen data.

The essence of the method is to build multiple trees in randomly selected subspaces of the feature space. Trees in different subspaces generalize their classification in complementary ways, and their combined classification can be monotonically improved.

In either method the stopping rule is until all the terminal nodes (leaves) contain points of a single class, or until it is impossible to split further. Since we do not want to lose any accuracy on classifying the training data, we do not consider methods to prune back the tree.

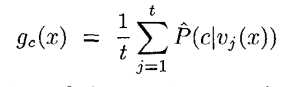
Our method to create multiple trees is to construct trees in randomly selected subspaces of the feature space. For a given feature space of m dimensions, there are 2" subspaces in which a decision tree can be constructed.

A decision tree is constructed in each selected subspace using the entire training set and the algorithms given in the previous section. Notice that each of these trees classifies the training data 100% correctly. Thus each tree generalizes its classification in a different way.

**Discriminant function**

For a point 2, let v3(x) be the terminal node that x is assigned to when it descends down tree T3 (j = 1,2, ..., t). Given this, let the posterior probability that x belongs to class c (c = 1,2, ..., n) be denoted by P(ClU3 (x)).

The discriminant function is defined as :



the decision rule is to assign x to class c for which gc(x) is the maximum.

**Single Trees**

First we show the results when single trees are constructed in the full feature space, as in the conventional practice. We tested both vectors fi and f2 and both of the tree growing methods.

Two heuristics used & compared: Central axis projection vs Perceptron training

**Random Forest**

F1 has 400 components. F2 has 852 components

The subspaces were restricted to 100 or 200 dimensions in the experiments, and the resultant differences in classification accuracy are clear from the figures.

## Shape Quantization and Recognition with Randomized Trees

*Yali Amit and Donald Geman [2]*

* Letter Recognition
* Looks Well Explained

## Random Forests,

*LEO BREIMAN, Statistics Department, University of California, Berkeley[3]*

* Long and mathematical

## Letter Recognition Using Holland-Style Adaptive Classifiers,

*Frey & Slate[4]*

**Originator of dataset paper (decision tree).**

# Comparison of Algorithms

## Approximate Statistical Tests for Comparing Supervised Classication Learning Algorithms Thomas G. Dietterich tgd@cs.orst.edu Department of Computer Science Oregon State University

**Explains pitfalls of type 1 error (noise). 10-fold cross validation.**

This paper reviews five approximate statistical tests for determining whether one learning algorithm out-performs another on a particular learning task. These tests are compared experimentally to determine their probability of incorrectly detecting a difference when no difference exists (type I error).

* Two widely-used statistical tests are shown to have high probability of Type I error in certain situations and should never be used. These tests are (a) a test for the difference of two proportions and (b) a paired-differences t test based on taking several random train/test splits.
* A third test, a paired-differences t test based on 10-fold cross-validation, exhibits somewhat elevated probability of Type I error.
* A fourth test, McNemar's test, is shown to have low Type I error.
* The fifth test is a new test, 5x2cv, based on 5 iterations of 2-fold cross-validation. Experiments show that this test also has acceptable Type I error.

# Algorithm Evaluation

## Ensemble methods based on bias–variance analysis

*Giorgio Valentini, Dipartimento di Informatica e Scienze dell’Informazione*

**Uses SVM’s so not sure**

bias–variance analysis is applied as a tool to explain the properties of ensembles of learners. A bias–variance analysis of ensembles based on resampling techniques is conducted, showing that, as expected, bagging is a variance reduction ensemble method,

## Mixtures of Conditional Maximum Entropy Models

*Dmitry Pavlov (Yahoo inc.), Alexandrin Popescul ; University of Pennsylvania, David M. Pennock Overture Services Inc; Lyle H. Ungar, ; University of Pennsylvania*

**Not sure why I picked this paper, uses Maximum Entropy to analyze results of algorithms.**

# ML Papers

## An Introduction to Variable and Feature Selection[5]

*Isabelle Guyon, Andre Elisseeff*

Perfectly correlated variables are truly redundant in the sense that no additional information is gained by adding them.

Very high variable correlation (or anti-correlation) does not mean absence of variable complementarity.

References

[1] Tin Kam Ho, “Random decision forests,” in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, Montreal, Que., Canada, 1995, vol. 1, pp. 278–282, doi: 10.1109/ICDAR.1995.598994.

[2] Y. Amit and D. Geman, “Shape Quantization and Recognition with Randomized Trees,” *Neural Comput.*, vol. 9, no. 7, pp. 1545–1588, Oct. 1997, doi: 10.1162/neco.1997.9.7.1545.

[3] L. Breiman, “Random Forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[4] P. W. Frey and D. J. Slate, “Letter recognition using Holland-style adaptive classifiers,” *Mach. Learn.*, vol. 6, no. 2, pp. 161–182, Mar. 1991, doi: 10.1007/BF00114162.

[5] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection,” *J. Mach. Learn. Res.*, vol. 3, p. 26, 2003.